Environmental Features Recognition for Lower Limb Prostheses Toward Predictive Walking

Kuangen Zhang, Caihua Xiong, Wen Zhang, Haiyuan Liu, Daoyuan Lai, Yiming Rong, Chenglong Fu

***Abstract*— This study aims to present a robust environmental features recognition system (EFRS) for lower limb prosthesis, which can assist the control of prosthesis by predicting locomotion modes of amputees and estimating environmental features in the following steps. A depth sensor and an inertial measurement unit (IMU) are combined to stabilize the point cloud of environments. Subsequently, the 2D point cloud is extracted from origin 3D point cloud and is classified through a neural network. Environmental features, including slope of road, width, and height of stair, were also estimated via the 2D point cloud. Finally, EFRS is evaluated through classifying and recognizing five kinds of common environments in simulation, indoor experiments and outdoor experiments by six healthy subjects and three transfemoral amputees, and databases of five healthy subjects and three amputees are used to validate without training. The classification accuracy of five kinds of common environments reach up to 99.3% and 98.5% for the amputees in the indoor and outdoor experiments, respectively. The locomotion modes are predicted at least 0.6 s before the switch of actual locomotion modes. Most estimation errors of indoor and outdoor environments features are lower than 5% and 10%, respectively. The overall process of EFRS takes less than 0.023 s. The promising results demonstrate the robustness and the potential application of the presented EFRS to help control of lower limb prostheses.**

***Index Terms*— prostheses, locomotion modes prediction, predictive walking, environmental features recognition, vision.**

I. Introduction

IT is predicted that the number of persons with lower limb amputations in the USA will increase to 3.6 million by the year of 2050 [1]. Without healthy limbs, disabled people will be inconvenient in daily life, especially for individuals with lower limb amputations. The need to help people with lower limb amputations to walk conveniently has inspired the development of powered lower limb prostheses [2]-[6].

A well-developed control approach of powered lower limb prosthesis is a finite-state controller, which decomposes gait into a series of distinct phases and implements a discrete set of parametric control [7]. Prostheses need to switch control strategies correctly in different locomotion modes [8]-[11]. However, for some commercially available powered lower limb prostheses, amputees need to instruct the prostheses their motion intent directly by pushing control buttons or executing an abnormal body motion. Consequently, the seamless transition of locomotion modes needs to be achieved by recognizing locomotion modes of amputees in real time [12].

Traditionally, electromyography (EMG) recorded from the residual limb, inertial measurement units (IMUs) and other mechanical sensors are used to recognize locomotion modes of amputees [13]. Myoelectric control algorithms have been used to detect amputees' locomotion modes [14]. EMG signal and forces/moments sensors have also been combined with a phase-dependent pattern classifier for continuous locomotion-mode identification [15]-[16]. Walking phases can be detected through IMU and other mechanical sensors [17]-[19]. Based on the EMG signal, some researchers can also realize proprioception and direct neural control of powered prostheses [20]. The recognition algorithms using signals from IMU and EMG may achieve high accuracy in one experiment, but the repeatability will be influenced by wear position and amputees' body condition [21]. Moreover, IMU and EMG signals are usually delayed, and cannot be used to detect environment and predict locomotion modes of amputees in advance. Therefore, the user-independent sensor could be a good choice to aid in prostheses to predict amputees' locomotion modes robustly.

Vision sensors are typical user-independent sensors, which will not be influenced by individual amputee' body condition. Furthermore, vision sensors can detect environments around amputees to provide contextual information and impose constraints on the movement possibility. A previous study provided a clear link between the vision and locomotion that the acquisition of the environment and self-motion will guide movements of the lower limbs to the appropriate landing targets [22]. Vision sensors are eyes of prostheses and can provide environmental information for prostheses to plan motion in the following steps. There are many methods to detect the obstacle and stairs using vision sensors [23]-[26]. Moreover, vision sensors have already been used in the humanoid robots [27]-[28].

Although vision sensors are commonly used in the humanoid robot, auto drive, and automatic guided vehicle fields, there are only a few researches about the prostheses with vision sensors. An IMU and a laser distance meter have been used in the terrain recognition module, which helped prostheses to recognize terrain accurately and transit locomotion mode smoothly [29]. While, the single point signal of the laser distance meter may be not enough to estimate some important features of the environment, such as the width and height of stairs, and the recognition algorithm

This work was supported in part by National Natural Science Foundation of China under Grant U1613206, 91648203 and Grant 51335004, and in part by Guangdong Innovative and Entrepreneurial Research Team Program under Grant 2016ZT06G587.

K. Zhang, W. Zhang, H. Liu, D. Lai, Y. Rong and C. Fu are with the Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen 518055, China (Corresponding author: Chenglong Fu: fucl@sustc.edu.cn).

C. Xiong is with the State Key Lab of Digital Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China.

could be invalid when the amputees are stationary. Moreover, the laser distance meter was positioned on the waist of amputees, which could be inconvenient for amputees. Using a SVM classifier, depth images can also be classified into different categories [30]. However, the level ground and ramps cannot be classified without the information of IMU.

Besides locomotion modes classification, environmental features estimation is also important for the control of prostheses. When amputees walk in various environments, estimation of environmental features is significant to optimize parameters of prostheses control. For instance, if amputees walk on the road, then the slope angle estimation of the road is necessary. If they walk on stairs, then the width and height estimation for one step of stairs is also useful. A Kinect was used for segmentation of stairs and features estimation [31], but recognition of other types of environments, such as slope, was not presented. Furthermore, the size of the Kinect was too large to be wearable.

Consequently, there are still many challenges to combine vision sensors with prostheses. Firstly, the precision of locomotion modes classification system presented above has not been evaluated in the outdoor environment. Secondly, the classification and recognition system should be fast and robust enough to ensure the stability of the prosthesis. Thirdly, vision sensors should be fixed on the self-contained lower limb prosthesis, but the motion of prosthesis will cause images of vision sensors to be unstable. These challenges inspire the research in this paper.

This study aimed to present a robust environmental features recognition system (EFRS) for self-contained lower limb prosthesis toward predictive walking. Major contributions of this study included 1) the robust locomotion modes prediction for seamless switch of prostheses' control strategies, 2) the accurate environmental features estimation for optimization of prostheses' control, 3) combination of vision sensor, IMU, and self-contained lower limb prostheses, 4) experimental verification for the indoor and outdoor experiments. The methodology of this study would be beneficial to increase the prediction accuracy of amputees' locomotion modes and optimize prostheses' control parameters in different environments. Eventually, the interaction between human and prostheses can be improved.

The rest of the paper is organized as follows. Section II describes methods of EFRS. The simulation and experimental methods are stated in section III and section IV respectively. Section V provides the experimental results and they are discussed in section VI. The conclusion of this paper is presented in section VII.

II. Methods

The environmental features recognition system (EFRS) proposed in this paper has two functions: subjects' locomotion modes prediction and environmental features estimation. The overall process of EFRS is illustrated in Fig. 1. As the first step, synchronized environmental point cloud and camera orientation information can be captured from the depth camera and IMU. Next, point cloud data can be offset and the dimension of point cloud can be reduced to generate some binary images. Consequently, categories of environment can be classified using a deep Convolutional Neural Network (CNN). According to classification results, features of the environment will be estimated, such as width and height for one step of stairs, and slope angle of the road. In the remainder of this section, detailed algorithms for EFRS will be presented.

A. Configuration of EFRS

In this study, a depth camera (CamBoard pico flexx, $68\text{mm} \times 17\text{mm} \times 7.25\text{mm}$, pmdtechnologies, Germany) and an IMU (MTi 1-series, $12.1\text{mm} \times 12.1\text{mm} \times 2.55\text{mm}$, Xsens, Netherlands) were used to recognize walking environment in front of the subjects. The experimental data acquisition and analysis were conducted using Matlab R2017b, an Intel Core i7-7700K (4.20GHz), 16 GB DDR3, and one graphics card: Quadro P400.

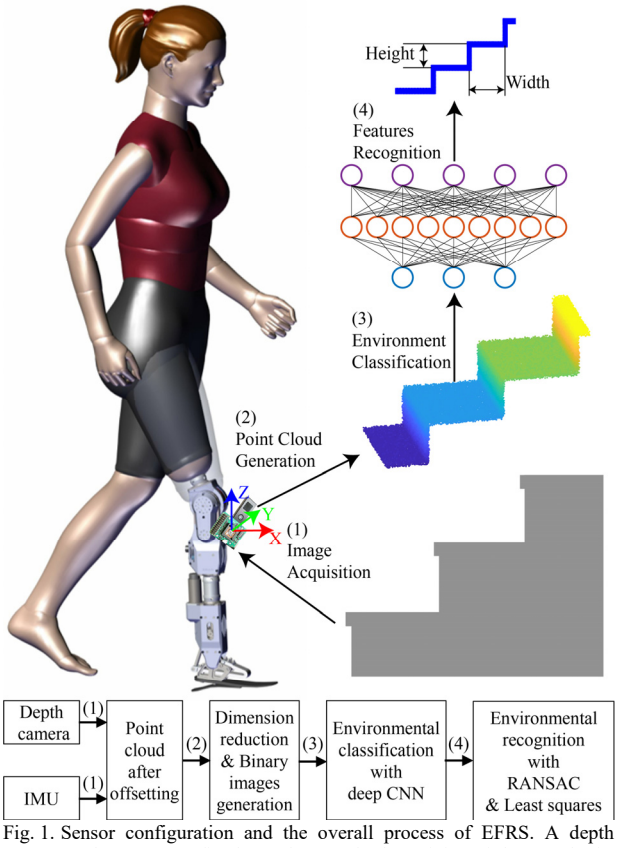

Fig. 1. Sensor configuration and the overall process of EFRS. A depth camera and IMU were fixed together on the knee joint of the prosthesis. Classification and recognition process of stairs were used to show one example of the overall process of EFRS.

The sensor configuration of IMU and depth camera is shown in Fig. 1. IMU and depth camera were fixed together as IMU-Camera sensor in order to combine vision information with camera orientation information. The data acquisitions of IMU and depth camera were running in two threads, and latest data of these two threads were captured to synchronize the IMU and camera data approximately. The depth camera can measure the 3D position of object points (point cloud) in the camera coordinate system. The IMU measured the orientation, angular velocity, and acceleration of the IMU-Camera sensor, which was fixed on the knee joint of the prosthesis. Although placing the IMU-Camera sensor on the knee resulted in large orientation variation during walking, the fusion of the orientation information and point

cloud can solve this problem.

B. Point Cloud Rotation Offset

The leg of human can swing about 60° within a gait cycle, which resulted in large orientation variation of the IMU-Camera. The output point cloud of depth camera was in the camera coordinate system. When the camera coordinate system changed, point cloud for the same object will also change. This phenomenon brings a difficulty to recognize the environment correctly. As shown in Fig. 2, the point cloud of level ground became an inclined plane rather than a horizontal plane in camera's view. Camera coordinate system changed when the camera rotated, but the ground coordinate system remained the same. In order to recognize environments correctly, the reference coordinate system of point cloud should be ground coordinate system rather than the camera coordinate system.

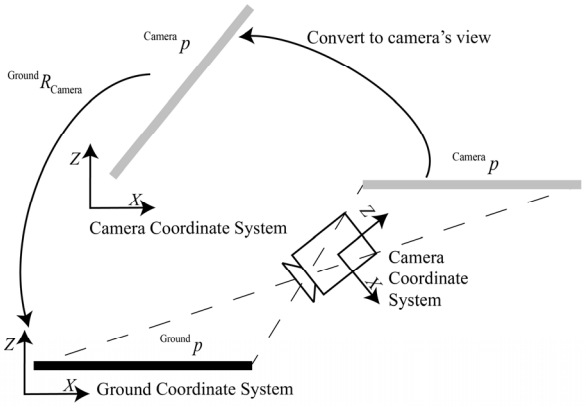

Fig. 2. Point cloud of level ground in different coordinate systems. ${}^{\text{Ground}}p$ and ${}^{\text{Camera}}p$ represent point cloud in the ground coordinate system and camera coordinate system respectively, and ${}^{\text{Ground}}R_{\text{Camera}}$ was the rotation matrix from the camera coordinate system to the ground coordinate system.

The rotation offset of point cloud can be realized by changing the reference coordinate system as:

$$ {}^{\text{Ground}}p = {}^{\text{Ground}}R_{\text{Camera}}\ {}^{\text{Camera}}p \tag{1} $$

where ${}^{\text{Ground}}p$ and ${}^{\text{Camera}}p$ were the point cloud in the ground coordinate system and camera coordinate system respectively, and ${}^{\text{Ground}}R_{\text{Camera}}$ was the rotation matrix from camera coordinate system to ground coordinate system, which was calculated by the software of IMU vendor, Xsens Technologies.

C. Dimension Reduction

The point cloud captured by depth camera was three-dimensional, but the categories and features of common environments illustrated in this paper can be estimated through the side view. People usually walk straightly, and if they turn around, the radius of turning is generally large. Therefore, the side view of the point cloud is usually enough to provide essential information for classification and features estimation of environments. After dimension reduction, the information changed from 3D to 2D, which remarkably reduced computation for classification and features estimation of different environments.

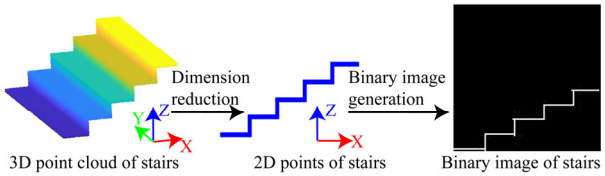

Fig. 3. Dimension reduction and binary images generation. 3D point cloud was made up of abundant points with 3D coordinates: X, Y, and Z. 2D point cloud were made up of fewer points with 2D coordinates: X and Z. Binary images only had two possible values: one or zero.

As shown in Fig. 3, 2D point cloud was extracted from 3D point cloud to reduce the dimension of point cloud

$$ \begin{cases} P_{3D} = \{(x_i, y_i, z_i) | i = 1, ..., n\} \\ K = \{i | -0.1\text{m} < y_i < 0.1\text{m}\} \\ P_{2D} = \{(x_k, z_k) | k \in K\} \end{cases} \tag{2} $$

where P_{3D} and P_{2D} were 3D point cloud and 2D point cloud respectively. x_i, y_i, and z_i were the coordinate values (unit: meter). K was the index set of extractive points from 3D points.

D. Binary Images Generation

A classic classification problem is a handwritten digit classification problem. The handwritten digit was pre-processed into a binary image, then the categories of these binary images can be classified through many methods [32]. The environments classification problem in this paper was similar to the problem of handwritten digit classification. The binary image of 100×100 pixels could be generated from 2D point cloud within 1m×1m based on (3), and one pixel in the binary image corresponded to one centimeter in the real world.

$$ \begin{aligned} & X = \{x_1, \ldots, x_k, \ldots, x_m\}, Z = \{z_1, \ldots, z_k, \ldots, z_m\} \\ & (x_{\min}, z_{\min}) = (\min\{x_k | x_k \in X\}, \min\{z_k | z_k \in Z\}) \\ & img = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 \end{bmatrix}_{100\times100} \\ & img(r,c) = \begin{cases} & 0.01(c-1) \le x_k - x_{\min} < 0.01c \\ 1, & \& \ 0.01(r-1) \le z_k - z_{\min} < 0.01r, \\ & x_k \in X, z_k \in Z \\ 0, & \text{otherwise} \end{cases} \end{aligned} \tag{3} $$

where X and Z were the set of x coordinates and y coordinates, and point (x_k, z_k) belonged to the 2D point cloud, and img was the binary image, and r and c was the number of row and column.

Firstly, a 2D grid was created to visualize the position distribution of 2D points in the X-Z coordinate plane. Secondly, a value representing the occupancy status of each

cell was generated. If there was no environment point in the cell, the occupancy status of this cell is false with a binary digit of 0, otherwise, the occupancy status of this cell is true with a binary digit of 1. Finally, binary occupancy images were generated based on this occupancy grid, which is shown in Fig. 3.

E. Classification Based on Neural Network

Image classification was realized by a deep CNN [33]. The CNN was selected to classify the images because it can extract deep features automatically rather than design complex features manually. Additionally, the convolution method can extract the local features of images, which can be used to classify images accurately. Moreover, the parameters sharing method in the convolution layer and down-sampling method in the max-pooling layer can decrease the number of required parameters and help the network to learn effectively.

The Neural Network Toolbox of Matlab was used to make a classification. As shown in Fig. 4, a network with three convolutional layers was designed. This convolutional neural network included an image input layer, three convolutional layers, three batch normalization layers, three Relu layers, two max-pooling layers, a fully connected layer, a softmax layer, and a classification layer.

The initial values of the weights of the convolutional and fully connected layers were randomly generated from a Gaussian distribution with mean 0 and standard deviation of 0.01. The initial biases were equal to 0. The batch normalization layers used an array of all zeros as the initial values.

In the convolutional layer, the size of the filter was set as 3×3 pixels and the number of filters was 16, 32, and 64 in the first, second, and third convolutional layer respectively. The padding was added to ensure that the spatial output size was the same as the input size. In the max pooling layer, the size of the rectangular region was set as 2×2 pixels, and the step size of training was 2. In the fully connected layer, the output size was set as 5.

After defining the network structure, training options were specified, and the maximum number of epochs, initial learning rate, and momentum value were set as 3, 0.001, and 0.8 respectively. The training dataset included 7500 images and each type of environments had 1500 images. The ratio of simulated images to overall training dataset was 50%. Because of the small size of training dataset, it only took 150 s to train the network.

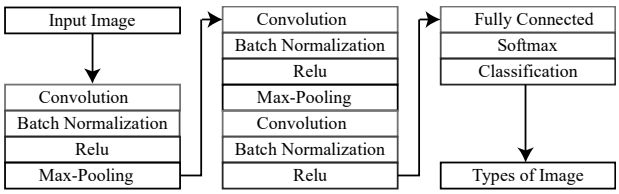

Fig. 4. The architecture of presented deep CNN. The CNN takes a binary image of 100×100 pixels as input. The output is a classification result of the input image for 5 classes. The convolutional layer convolves the input by moving the filters to compute the dot product of the input and the weights and then adding a bias term. Batch normalization layers normalize each input channel across a mini-batch. In a Relu layer, each element of the input which is less than zero is set to zero. Down-sampling is performed in a max pooling layer by dividing the input into rectangular regions and computing the maximum of each region. A fully connected layer multiplies the input by a weight matrix and then adds a bias vector. A softmax layer applies a softmax function to the input to calculate probabilities. Classification layer uses the probabilities for each input to assign the input to one of the mutually exclusive classes and compute the loss.

F. Classification Results Filtering

In this work, a median filter was implemented to decrease error points of classification results. The filter smoothed classification results by using a sliding window, and each entry was replaced with the median of this window. Because classification result was a 1D signal, the sliding window included a few preceding and following entries. Although this method led to the delay of transition, it can decrease the error rate of classification, which is shown in section V. In order to minimize delay and ensure camera to recognize the transition place before subject arrives at this transition place, the size of the sliding window was set as 7 in the indoor experiments, and as 11 in the outdoor experiments.

G. Environment Features Estimation

The features of the environment were estimated based on the classification results and extracted 2D point cloud of dimension reduction. If the environment was classified as a road, the slope angle θ will be estimated using a linear least squares fitting algorithm as:

$$\begin{cases} \overline{x}=\sum_{k=1}^{m}x_k\big/m,\quad \overline{z}=\sum_{k=1}^{m}z_k\big/m \\ \theta=\arctan\left(\dfrac{\sum_{k=1}^{m}(x_k-\overline{x})(z_k-\overline{z})}{\sum_{k=1}^{m}(x_k-\overline{x})^2}\right) \end{cases} \tag{4}$$

where point (x_k,z_k) belonged to the 2D point cloud. $(\overline{x},\overline{z})$ was the mean point of the 2D point cloud. m was the count of points in the 2D point cloud. θ was the slope angle of the road.

If the environment was classified as stairs, two median lines with most inliers were fitted based on the random sample consensus (RANSAC) algorithm:

$$\begin{cases} P_{\text{inliers1}}=\left\{(x_k,z_k)\middle|\left|z_k-z_{\text{inliers1}}\right|\le d\ \&\ (x_k,z_k)\in P_{\text{2D}}\right\} \\ z_{\text{inliers1}}=\arg\max\limits_{z_{\text{inliers1}}}\left(\left|P_{\text{inliers1}}\right|\right) \\ P_{\text{inliers2}}=\left\{\begin{matrix}(x_k,z_k)\middle|\left|z_k-z_{\text{inliers2}}\right|\le d\\ \&\ (x_k,z_k)\in P_{\text{2D}}-P_{\text{inliers1}}\end{matrix}\right\} \\ z_{\text{inliers2}}=\arg\max\limits_{z_{\text{inliers2}}}\left(\left|P_{\text{inliers2}}\right|\right) \\ h=\left|z_{\text{inliers2}}-z_{\text{inliers1}}\right| \\ x_{\text{1max}}=\max(\{x_l\middle|(x_l,z_l)\in P_{\text{inliers1}}\}) \\ x_{\text{2max}}=\max(\{x_j\middle|(x_j,z_j)\in P_{\text{inliers2}}\}) \\ w=\left|x_{\text{2max}}-x_{\text{1max}}\right| \end{cases} \tag{5}$$

where P_{2D} was 2D point cloud, which was introduced in (2). P_{inliers1} and P_{inliers2} were sets of inliers in the first fitting and the second fitting respectively. The parameter d was the threshold of distance. $\left|P_{\text{inliers1}}\right|$ and $\left|P_{\text{inliers2}}\right|$ were the number

of elements in P_{inliers1} and P_{inliers2} respectively. Here w and h were the width and height of one step of stairs respectively.

As shown in Fig. 5 and (5), a horizontal line with the most inliers was fitted firstly. Then inliers for the first line were removed and another horizontal line with the most inliers was fitted secondly. Subsequently, the median z coordinates of inliers for two lines were calculated, and the distance between these median lines was estimated as the height of stair. Finally, the maximum x coordinates of the inliers for two lines were also calculated, and the distance between these maximum x coordinates was estimated as the width of the stair.

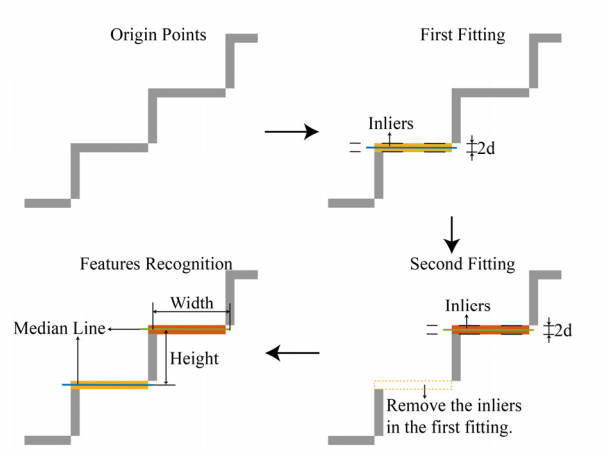

Fig. 5. Estimation methods of stairs. d was the threshold of distance and was set as 0.1 m in this paper.

III. Simulation

In order to verify the effectiveness of the proposed method, simulations were carried out. In the simulations, the images of five different types of common environments were generated as shown in Fig. 6. These environments included up stairs, down stairs, up ramp, down ramp, and level ground. These images were divided into two parts: training set (60%) and validation set (40%). The images in the training set were used to train the classification network, and the images in the validation set were used to validate the accuracy of the classification results. The environmental features were also estimated, and estimated features were compared with actual features to verify the precision of the EFRS.

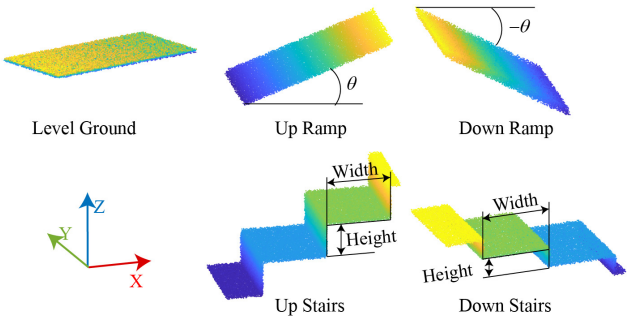

Fig. 6. Five types of environments were generated in the simulation. The features need to be estimated were: slope angle of ramp, width, and height of stairs.

A. Simulation Database Generation

The point cloud was generated according to the characteristics of up stairs, down stairs, up ramp, down ramp, and level ground. The geometries of these five types of environments were defined in (6)-(9).

Level ground: point cloud of level ground was a horizontal plane, which can be defined as:

$$0 \le x \le 1, 0 \le y \le 0.5, z = 0.01 \cdot \text{rand} \tag{6}$$

where x, y, z were the coordinate values of points in the point cloud. The function of rand was the uniformly distributed random number in the interval (-1,1), which was used to introduce noise in the point cloud and validate the robustness of the presented method.

Up Stairs: point cloud of up stairs was made up of several horizontal and vertical planes, which can be defined as:

$$\begin{cases} w = 0.32 + 0.05 \cdot \sin(t), \quad h = 0.15 + 0.05 \cdot \sin(t) \\ 0 \le y \le 0.5, \quad t \in \{1, 2, \ldots, 1000\} \\ noise = 0.01 \cdot \text{rand} \end{cases}$$

$$z = \begin{cases} noise & , 0 \le x \le 0.2 \\ h + noise & , 0.2 \le x \le 0.2 + w \\ 2h + noise & , 0.2 + w \le x \le 0.2 + 2w \\ 3h + noise & , 0.2 + 2w \le x \le 1 \end{cases} \tag{7}$$

$$x = \begin{cases} 0.21 + noise & , 0.01 \le z \le h - 0.01, h > 0 \\ 0.21 + w + noise & , h + 0.01 \le z \le 2h - 0.01, h > 0 \\ 0.21 + 2w + noise & , 2h + 0.01 \le z \le 3h - 0.01, h > 0 \end{cases}$$

where w was the width of the stair, h was the height of stair, and t was the index of the point cloud.

Down Stairs: point cloud of down stairs was define as:

$$\begin{cases} w = 0.32 + 0.05 \cdot \sin(t), \quad h = -0.15 + 0.05 \cdot \sin(t) \\ 0 \le y \le 0.5, \quad t \in \{1, 2, \ldots, 1000\} \\ noise = 0.01 \cdot \text{rand} \end{cases}$$

$$z = \begin{cases} noise & , 0 \le x \le 0.2 \\ h + noise & , 0.2 \le x \le 0.2 + w \\ 2h + noise & , 0.2 + w \le x \le 0.2 + 2w \\ 3h + noise & , 0.2 + 2w \le x \le 1 \end{cases} \tag{8}$$

$$x = \begin{cases} 0.19 + noise & , h + 0.01 \le z \le -0.01, h < 0 \\ 0.19 + w + noise & , 2h + 0.01 \le z \le h - 0.01, h < 0 \\ 0.19 + 2w + noise & , 3h + 0.01 \le z \le 2h - 0.01, h < 0 \end{cases}$$

where w was the width of the stair, and h was the height of stair, and t was the index of the point cloud.

Ramp: point cloud of ramp can be defined as:

$$0 \le x \le 1, \quad 0 \le y \le 0.5, \quad z = \tan(\theta) \cdot x + 0.01 \cdot \text{rand}$$

$$t \in \{1, 2, \ldots, 1000\}, \theta = \begin{cases} 15 + 5 \cdot \sin(t) & , \text{Up Ramp} \\ -15 + 5 \cdot \sin(t) & , \text{Down Ramp} \end{cases} \tag{9}$$

where θ was the slope angle of the ramp, and t was the

index of the point cloud.

As defined in (6)-(9), environmental features were simulated as sinusoids to change values of environmental features and validate the accuracy of environmental features estimation. After generating the point cloud of 5 types of environments, the binary images were generated based on the methods presented in section II.D, as shown in Fig. 7.

B. Environment Classification Test

One important purpose of EFRS was to predict different locomotion modes accurately. The environment classification accuracy was defined as the percentage of correct classification results out of the total number of classification results. The classification result was considered as correct if the output result of EFRS was the same as the type of environment of the input point cloud. The database of simulation was divided into training set (60%) and validation set (40%) to evaluate the classification accuracy of EFRS.

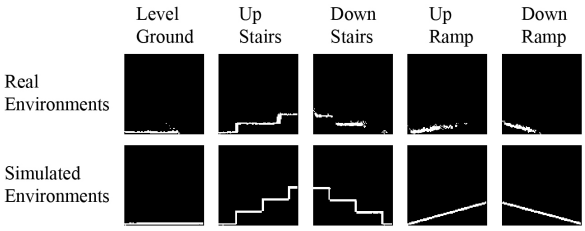

Fig. 7. Binary images of five types of environments. Each column of binary images represents the same category of environments. The binary images in the upper row are generated in real experiments, and which in the lower row are generated in the simulation.

C. Environmental Features Estimation Test

Another important purpose of EFRS was to estimate environmental features. The error of environmental features estimation was defined as the deviation between the estimation parameters of features and actual parameters of features. As can be observed from (7)-(9), features were designed as specific curves. The error of environmental features estimation can be analyzed by comparing output curves of features and designed curves of features.

IV. Experimental Methods

A. Subjects Information

Six healthy subjects and three transfemoral amputees were invited to experiments. Amputees were recruited from the local prosthetic company, and two healthy subjects were authors of this paper, and other healthy subjects were invited from Southern University of Science and Technology. The anthropomorphic data of the above subjects and configuration information of prosthesis for three amputees are shown in TABLE I and TABLE II, respectively. The approval to perform these experiments was granted by the Review Board of Southern University of Science and Technology, and informed consents were obtained for subjects prior to the experiments.

B. Experimental Setup

Indoor and outdoor experiments were conducted respectively. As shown in Fig. 8 and Fig. 9, there were five different types of experiments: level ground, up stairs, down stairs, up ramp, and down ramp. All subjects and amputees were requested to walk in indoor and outdoor environments repeatedly five times. In each trial, there are 1 up and down stairs locomotion modes, 1 up and down slope locomotion modes, and 3 level ground locomotion modes.

TABLE I
Basic Information of Healthy Subjects

Subjects	Height (m)	Weight (kg)	Age (years)	Gender
Subject 0	1.73	63	23	Male
Subject 1	1.66	59	28	Male
Subject 2	1.65	63	30	Male
Subject 3	1.68	58	29	Male
Subject 4	1.72	60	24	Female
Subject 5	1.67	53	25	Male

TABLE II
Basic Information of Amputees

Subjects	Amputee 1	Amputee 2	Amputee 3
Height (m)	1.70	1.70	1.69
Weight (kg)	64	60	62
Age (years)	38	38	42
Gender	Male	Male	Male
Amputation time	2016	2001	2000
Amputation side	Right	Left	Left
Residual limb length (m)	0.33	0.30	0.31

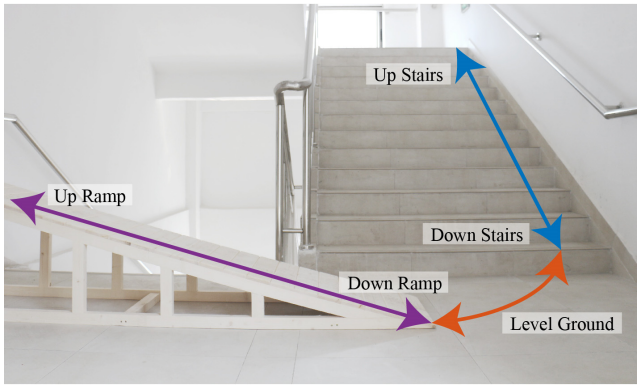

Fig. 8. Indoor experimental environment.

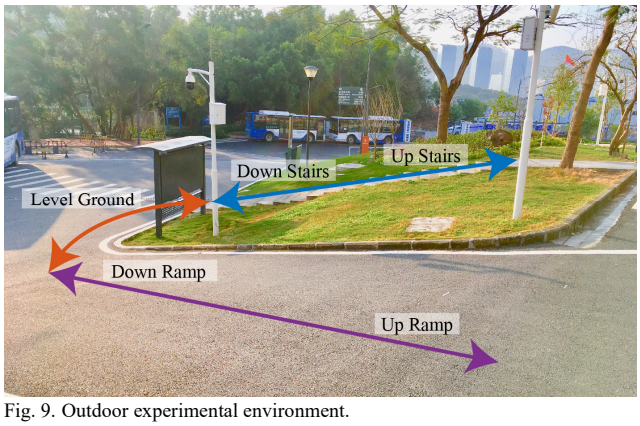

Fig. 9. Outdoor experimental environment.

Because the binary images generated in the real experiments may not be as complete and clear as simulation data, the real data should be added to the training set to increase robustness. Additionally, unlike the IMU or EMG data, the vision data is independent of subjects. Therefore, we only added the real data of one subject to the training set to

validate the user-independence of this system. The database of one healthy subject (subject 0) was combined with a simulation database and was divided into training set (60%) and validation set (40%) to train the network firstly. After finishing the training of network, databases of other five healthy subjects (subject 1-5) and three amputees (amputee 1-3) were used to validate (100%) without training.

As shown in Fig. 10, the EFRS was positioned on the upper patellar tendon of subjects to collect depth images of environment and orientation information of camera. The resolution of depth image was 224×171 pixels, and depth images were captured at 25 frames per second in the indoor experiment and 15 frames per second in the outdoor experiment. In order to evaluate the accuracy of EFRS, a RGB camera was used to focus on the walking environment around the foot to record current locomotion mode. Video streams of the depth camera and those of RGB camera were synchronized to compare the prediction labels and actual labels. Based on the above video streams, correct modes were labeled manually to validate the accuracy of EFRS.

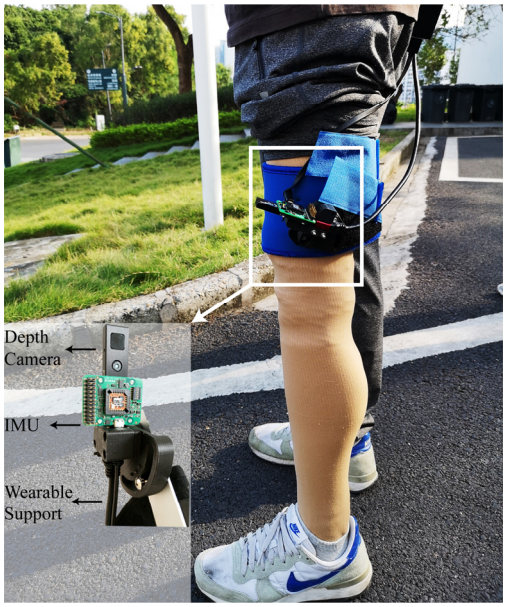

Fig. 10. Experimental setup on an amputee. Depth camera and IMU were fixed on a wearable support. The wearable support was attached on the upper patellar tendon of subjects

C. Environment Classification Test

The environmental classification test was based on the database of subject 1-5 and amputee 1-3 without training. Classification results were compared with correct modes (labeled manually) to evaluate the accuracy of classification.

D. Environmental Features Estimation Test

The environmental features estimation test was based on the database of subject 1-5 and amputee 1-3 without training. The slope angle of the ramp and the width and height of stair were measured before experiments. Values of environmental features were estimated based on EFRS, and errors between estimation values and actual values were evaluated.

V. Results

A. Convergence of Neural Network

The convergence of presented neural network for the overall dataset is shown in Fig. 11. The ratio of simulated images to overall training dataset was 1:2, and remaining images were generated from indoor and outdoor experiments of one subject (subject 0). The accuracy reached above 90% and the loss decreased to 0.2 after about 20 iterations. After 174 iterations, the accuracy reached 98.04% and loss decreased to about 0.1.

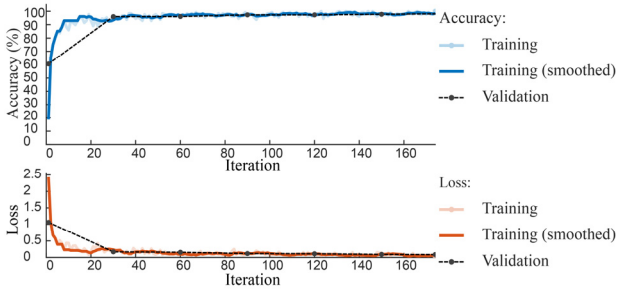

Fig. 11. Convergence of presented neural network. Training accuracy, smoothed training accuracy, and validation accuracy were classification accuracy on each individual mini-batch, its smoothed, and classification accuracy on the entire validation set respectively. Training loss, smoothed training loss, and validation loss were the loss on each mini-batch, its smoothed version, and the loss on the validation set, respectively.

B. Simulation Results

The classification performance of the EFRS was evaluated according to the classification experiment of the simulated point cloud. The classification of each sample took less than 3 ms and achieved a 100% accuracy. The high accuracy of classification shows the feasibility of EFRS to aid in the control of prosthesis. Furthermore, the accuracy of classification in the real experiment was also validated in the following experiments.

The environmental features estimation was also accurate, and the max percentage errors of slope angle for up ramp and down ramp were 0.5737% and 0.7102% respectively. The max percentage errors of width and height of stair were 0% and 1.45% respectively.

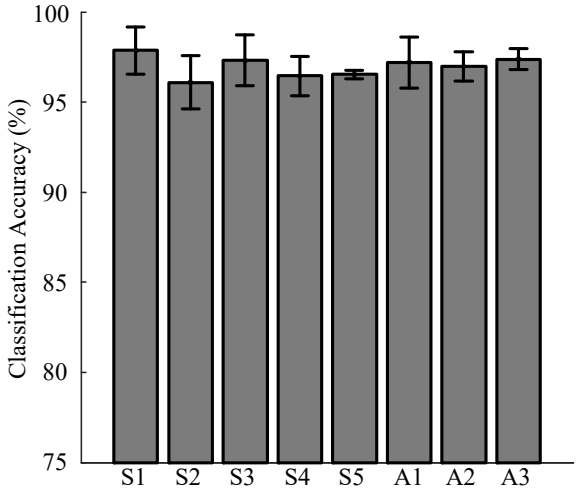

Fig. 12. Classification results of the indoor experiment of healthy subjects (S1-S5) and amputees (A1-A3). Error bars indicate mean values and standard deviations of classification accuracy in five repeating experiments.

TABLE III
INDOOR ENVIRONMENTAL FEATURES ESTIMATION ERRORS

Subject number	S1	S2	S3	S4	S5	A1	A3	A3
Slope angle of up ramp (14.7°)	7.6%	4.2%	8.4%	4.0%	4.4%	0.6%	0.5%	3.3%
Slope angle of down ramp (−14.7°)	5.9%	0.3%	1.5%	2.1%	0.1%	6.0%	3.7%	0.2%
Width of up stairs (0.3 m)	3.1%	4.6%	5.8%	2.8%	3.0%	1.0%	2.3%	1.6%
Height of up stairs (0.15 m)	1.4%	3.2%	1.7%	2.2%	1.2%	2.3%	0.5%	0.8%
Width of down stairs (0.3 m)	0.6%	2.0%	3.7%	0.0%	1.5%	3.3%	2.7%	1.4%
Height of down stairs (0.15 m)	1.9%	0.7%	1.0%	1.3%	0.8%	0.6%	2.7%	3.2%

C. *Indoor Experiment Results*

The classification performance of the EFRS was validated through the indoor classification experiment secondly. The classification of each sample took about 3 ms, and results of the classification accuracy of the filtered result are shown in Fig. 12. Mean values of classification accuracy of all subjects and amputees were higher than 95%, and the highest classification accuracy was 99.6% for healthy subjects and 99.3% for amputees.

In order to show classification results intuitively, classification result of one subject (subject 0) was shown in Fig. 13. The accurate definition and measurement of locomotion transition time are very difficult, because human intent is not an intuitive signal. Therefore, the previous researchers have to use some gait events to estimate the deadline of locomotion transition and change the control mode of the prosthesis, such as toe off from the level ground to up stairs and heel contact from down stairs to the level ground [16], [34]. If the vision system can recognize the change of environments in advance, then the gait event (toe off and heel contact) can be used to trigger the transition of locomotion mode. It is difficult to estimate the actual transition time of human intent accurately, but the classification of environments in advance and measurement of gait event may be enough to control the prosthesis to change locomotion mode smoothly.

The lead time is used to evaluate the performance of EFRS. The lead time is defined as the time elapse between the time when the vision system recognizes the change of environments and the deadline of locomotion transition for the prosthesis, which is similar to the TR response time in [29]. As can be seen from Fig. 13, the minimum lead time was 0.6 s, which meant that the vision system can detect the mode transition at least 0.6 s ahead of subjects' action.

As shown in Fig. 14, the accuracy of estimation in the indoor experiment is high, and estimation results can match the real values of these features. Estimation error is defined as the ratio of deviated value divided by the actual value. Estimation errors of environmental features estimation experiments are shown in TABLE III, and most estimation errors are lower than 5%.

The estimation of stair dimensions took less than 20 ms, and the slope angle estimation of each road sample took less than 1 ms. Therefore, the maximum processing time for the overall process of EFRS in the indoor experiment was 23 ms and the processing speed of EFRS can be as high as 40 frames/s. Considering the minimum lead time as 0.6 s, categories, and environmental features can be estimated at least 0.4 s ahead of subjects' action.

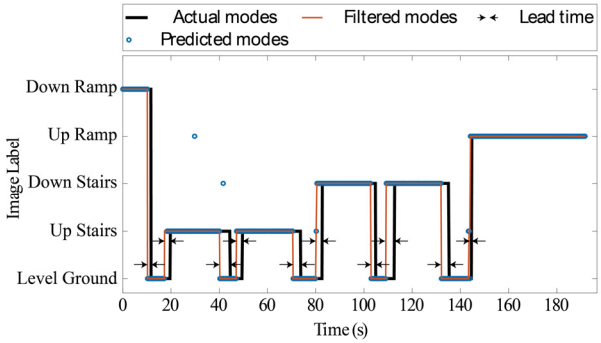

Fig. 13. Classification results of the indoor experiment. Predicted modes were output results of the neural network. Predicted modes were filtered by a median filter, and the filtered results were shown as filtered modes. Lead time is defined as the time elapse between the time when the vision system recognizes the change of environments and the deadline of locomotion transition for the prosthesis.

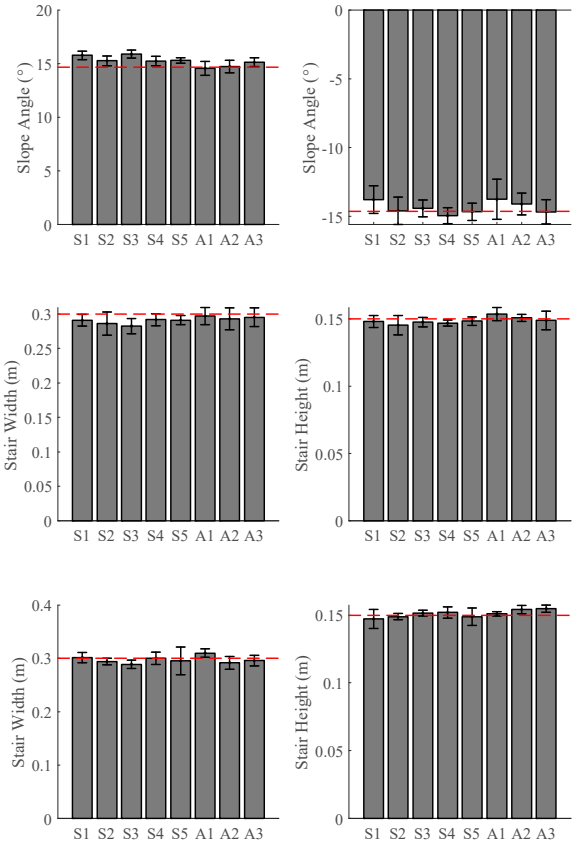

Fig. 14. Environmental features estimation results of the indoor experiment of five healthy subjects (S1-S5) and three amputees (A1-A3). Red dash lines represented actual values of features, and error bars represented mean values and standard deviations of estimated values.

TABLE IV
OUTDOOR ENVIRONMENTAL FEATURES ESTIMATION ERRORS

Subject number	S1	S2	S3	S4	S5	A1	A3	A3
Slope angle of up ramp (11°)	2.3%	4.6%	1.3%	3.7%	3.1%	4.7%	6.0%	9.5%
Slope angle of down ramp (−11°)	11.1%	6.4%	3.0%	4.4%	0.4%	4.6%	5.0%	9.0%
Width of up stairs (0.46 m)	2.6%	3.6%	13.7%	2.7%	5.9%	6.1%	1.2%	1.2%
Height of up stairs (0.11 m)	5.2%	1.2%	3.0%	0.5%	0.2%	2.4%	1.8%	9.2%
Width of down stairs (0.46 m)	8.9%	1.3%	3.1%	1.0%	3.3%	2.7%	12.9%	4.3%
Height of down stairs (0.11 m)	4.2%	1.6%	3.4%	6.6%	3.0%	1.4%	0.3%	1.9%

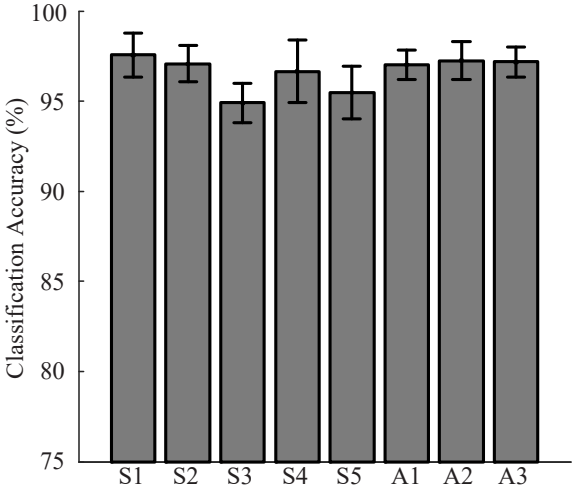

Fig. 15. Classification results of the outdoor experiment of healthy subjects (S1-S5) and amputees (A1-A3). Error bars indicate mean values and standard deviations of classification accuracy in five repeating experiments.

D. Outdoor Experiment Results

The performance of EFRS was validated through outdoor classification experiment finally. The classification of each sample took about 3 ms. As shown in Fig. 15, mean values of classification accuracy of all subjects and amputees were still higher than 94.9%, and the highest classification accuracy was 98.9% for healthy subjects and 98.5% for amputees.

Classification result of one subject (subject 0) was also shown in Fig. 16, the minimum lead time was 3.34 s, which indicates that the vision system can detect the mode transition at least 3.34 s ahead of subjects' action.

Environmental features estimation results are shown in Fig. 17. The accuracy of estimation in the outdoor experiment was also high, and estimation results can match the real values of these features. Estimation errors of the environmental features estimation experiments are shown in TABLE IV, and most estimation errors are lower than 10%.

The estimation of stair dimensions also took less than 20 ms, and the slope angle estimation of each road sample took less than 1 ms. Therefore, the maximum processing time for the overall process of EFRS in the outdoor experiment was 23 ms. Considering the minimum lead time was 3.34 s, categories, and environmental features can be predicted at least 3 s ahead of subjects' action.

Furthermore, the lead time can be adjusted by controlling the range of point cloud captured by the camera. For instance, the range of the point cloud can be limited into one stride, then the lead time would be suitable to guide the locomotion transition in the next step.

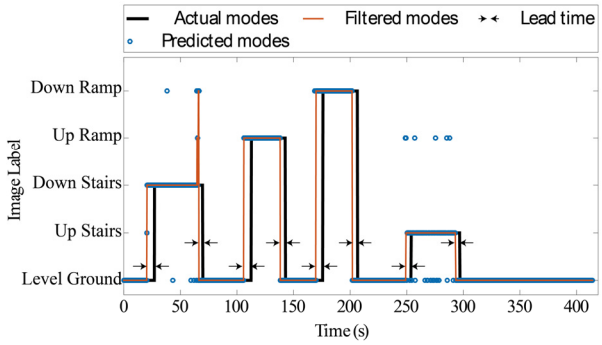

Fig. 16. Classification results of the outdoor experiment. The definitions of predicted modes, filtered modes, and lead time were as same as the definitions in Fig. 13.

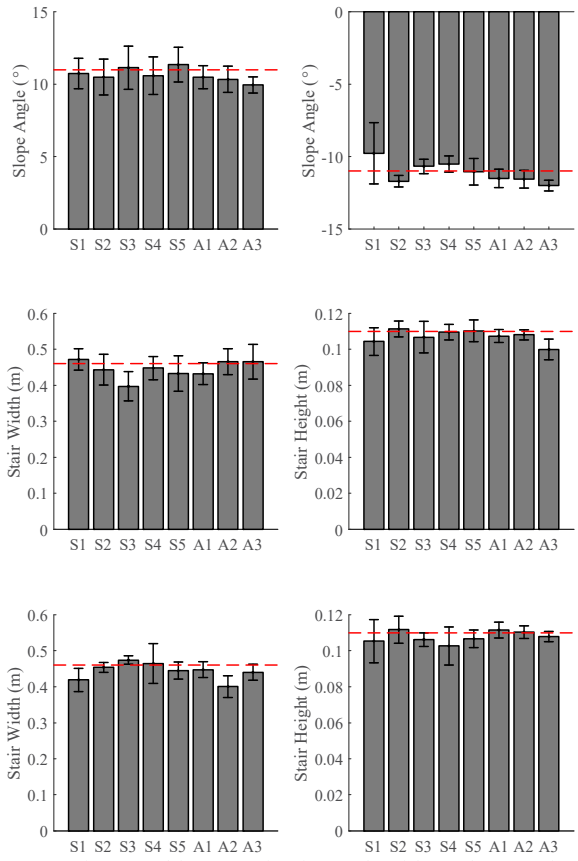

Fig. 17. Environmental features estimation results of the outdoor experiment of five healthy subjects (S1-S5) and three amputees (A1-A3). Red dash lines represented actual values of features, and error bars represented mean values and standard deviations of estimated values.

VI. Discussion

A. Advantages of the Proposed Method

This paper introduced a robust environmental features recognition system (EFRS) for lower limb prosthesis toward predictive walking. Several qualities of proposed EFRS might underlie its effectiveness in the predictive control of prosthesis.

Firstly, the precision of the presented classification algorithm was evaluated in the indoor and outdoor experiments, and experiments results suggested the robustness of the presented EFRS. The highest classification accuracy for presented classification algorithm in this paper can achieve 99.3% for amputees in the indoor experiments and 98.5% for amputees accuracy in the outdoor experiments, which were about 4% higher than the precision of SVM classifier introduced in [30]. The SVM classifier was used to classify depth images into different environmental patterns and reached an accuracy of 94.1% in the indoor environments classification experiments. An IMU and a laser distance meter were used in the terrain recognition module and they achieved an accuracy of 98% in the indoor experiments [29], but the recognition algorithm based on single point signal of laser distance meter could be invalid when the subjects were stationary. Therefore, depth camera could be a better choice and the presented EFRS in this paper could be more robust.

Moreover, the environmental features of five common types of environments were estimated accurately in this paper. Most errors of estimation results in this paper were less than 5% and were as low as the estimation errors presented in [31]. In their research, the point cloud provided by the Kinect was used for segmentation of stairs and features estimation and the estimation errors of stair width and stair height were about 4.1% and 5% respectively. While, Kinect was not wearable, and depth camera in this paper was fixed on the leg of the subject. Hence, the presented EFRS in this paper could be more practical to aid in control of prosthesis.

In addition to high accuracy of proposed EFRS, the speed of classification and recognition system in this proposed method was high. It took about 14 ms to classify an image based on a SVM classifier [30] and 200 ms to recognize a point cloud in [31]. In this paper, it took less than 3 ms (21% of 14 ms) to classify an image and about 20 ms (10% of 200 ms) to recognize a point cloud. Although the speed of the proposed algorithm was only evaluated on the computer with an Intel Core i7-7700K, the speed could remain high on the embedded systems after using FPGA and converting Matlab programs to C++ programs. Moreover, the algorithms in previous researches were also implemented on computers rather than embedded systems and their algorithms were time-consuming because 3D point cloud or depth images were processed directly. Conversely, the size of the data needs to be processed for each frame in this paper was decreased to 1 KB because of dimension reduction algorithm. Therefore, the speed of classification and recognition system was higher in this paper.

Furthermore, the vision sensor in this paper can be fixed on the self-contained prosthesis, and an IMU was utilized to stabilize point cloud of the vision sensor. The laser distance meter used in [29] was positioned on the waist of subjects to decrease noise caused by the motion of subjects, which could be inconvenient for subjects. In this paper, the depth sensor and IMU was positioned on the upper patellar tendon of a healthy subjects and can be positioned around knee of prostheses, which ensured that the vision system can be fixed in lower limb prostheses and assist subjects to feel more comfortable to use prostheses without the need to wear other wearable devices. Moreover, an IMU was used to measure the orientation of depth camera in real time to stabilize point cloud generated by the depth camera. The stable point cloud could be used to classify level ground and ramp, which was better than the vision system used in [30]. They did not use an IMU to provide orientation information of the camera, thus the level ground, up ramp, and down ramp cannot be classified.

TABLE V
Comparison with Previous Researches

Parameters	[29]	[30]	[31]	This Paper
Best Classification Accuracy	98% (indoor)	94.1% (indoor)	\	98.5% (outdoor)
Subject type	Amputee	Healthy subject	Healthy subject	Amputee
Environments Modes	5	3	1	5
Classification Speed	\	70 frames/s	\	330 frames/s
Self-contained or Not	No	Yes	\	Yes
Features Estimation Accuracy	\	\	5%	5%

As shown in TABLE III, the vision system presented in this paper had more functions because it can classify five types of common environments and estimate environmental features. The environmental features recognition vision system described in this paper reached higher classification and estimation accuracy with higher speed. Moreover, the vision system could be fixed on a self-contained prosthesis. Finally, the presented EFRS was evaluated by both healthy subjects and amputees and achieved high accuracy for all subjects and amputees, which demonstrates that the presented EFRS is user-independent.

B. Limitations and Future Works

Classification results from vision system only provided possible modes for the following steps, and there will be a temporal error between the classification results of EFRS and actual locomotion modes of subjects. Because human intent is not an intuitive signal and the accurate definition and measurement of locomotion transition time are very difficult, the gait event (toe off and heel contact) can be used to estimate the transition time and trigger the transition of locomotion mode. The gait events can be estimated by using IMU, EMG, or pressure sensors, which will be implemented in our future work.

Additionally, although the median filter can decrease the noise of classification results, the sliding window of the median filter will cause a time delay in online classification. Therefore, a decision optimization method should be designed to reduce the required window length and time delay.

Moreover, the classification and estimation results of EFRS were not used to guide the control of a real prosthesis

in this paper. Therefore, the EFRS should be applied on a real prosthesis to predict the locomotion modes transition online in the future.

VII. Conclusion

In this paper, a robust environmental features recognition system (EFRS) for lower limb prosthesis toward predictive walking was presented. The depth sensor and inertial measurement unit (IMU) could be fixed on the knee joint of the prosthesis to help prostheses detect environmental context. The designed vision system was evaluated using simulation, indoor experiments, and outdoor experiments by six healthy subjects and three amputees, and a database of five healthy subjects and three amputees were used to validate without training. The highest accuracy of classification results in simulation, indoor experiments, and outdoor experiments was 100%, 99.3% for amputees, and 98.5% for amputees respectively. The vision system could predict the mode transitions at least 3.34 s before subjects switched the locomotion modes in the outdoor experiment, and the minimum lead time in the indoor experiment was 0.6 s. Most errors of estimation results were lower than 5%, and the overall process of the presented vision system took less than 0.023 s. These promising results demonstrated the robustness and the potential application of the presented environmental features recognition vision system to assist the control of lower limb prostheses.

References

[1] K. Ziegler-Graham, E. J. MacKenzie, P. L. Ephraim, T. G. Travison, and R. Brookmeyer, "Estimating the prevalence of limb loss in the United States: 2005 to 2050," *Archives of Physical Medicine and Rehabilitation*, vol. 89, no. 3, pp. 422–429, Mar. 2008.

[2] B. Davies and D. Datta, "Mobility outcome following unilateral lower limb amputation," *Prosthetics and Orthotics International*, vol. 27, no. 3, pp. 186–190, Jan. 2003.

[3] A. M. Dollar and H. Herr, "Lower extremity exoskeletons and active orthoses: challenges and state-of-the-art," *IEEE Transactions on Robotics*, vol. 24, no. 1, pp. 144–158, Feb. 2008.

[4] R. J. Farris, H. A. Quintero, and M. Goldfarb, "Preliminary evaluation of a powered lower limb orthosis to aid walking in paraplegic individuals," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 19, no. 6, pp. 652–659, Dec. 2011.

[5] S. K. Au and H. M. Herr, "Powered ankle-foot prosthesis," *IEEE Robotics Automation Magazine*, vol. 15, no. 3, pp. 52–59, Sep. 2008.

[6] F. Sup, A. Bohara, and M. Goldfarb, "Design and control of a powered transfemoral prosthesis," *The International Journal of Robotics Research*, vol. 27, no. 2, pp. 263–273, Feb. 2008.

[7] M. R. Tucker *et al.*, "Control strategies for active lower extremity prosthetics and orthotics: a review," *Journal of NeuroEngineering and Rehabilitation*, vol. 12, no. 1, pp. 1-29, Jan. 2015.

[8] A. M. Simon *et al.*, "Configuring a powered knee and ankle prosthesis for transfemoral amputees within five specific ambulation modes," *PLOS ONE*, vol. 9, no. 6, p. e99387, Jun. 2014.

[9] E. D. Ledoux and M. Goldfarb, "Control and evaluation of a powered transfemoral prosthesis for stair ascent," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 7, pp. 917–924, Jul. 2017.

[10] S. Culver, H. Bartlett, A. Shultz, and M. Goldfarb, "A stair ascent and descent controller for a powered ankle prosthesis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 26, no. 5, pp. 993–1002, May 2018.

[11] K. Suzuki, G. Mito, H. Kawamoto, Y. Hasegawa, and Y. Sankai, "Intention-based walking support for paraplegia patients with Robot Suit HAL," *Advanced Robotics*, vol. 21, no. 12, pp. 1441–1469, Jan. 2007.

[12] H. A. Varol, F. Sup, and M. Goldfarb, "Multiclass real-time intent recognition of a powered lower limb prosthesis," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 3, pp. 542–551, Mar. 2010.

[13] D. Novak and R. Riener, "A survey of sensor fusion methods in wearable robotics," *Robotics and Autonomous Systems*, vol. 73, pp. 155–170, Nov. 2015.

[14] J. D. Miller, M. S. Beazer, and M. E. Hahn, "Myoelectric walking mode classification for transtibial amputees," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 10, pp. 2745–2750, Oct. 2013.

[15] H. Huang, T. A. Kuiken, and R. D. Lipschutz, "A strategy for identifying locomotion modes using surface electromyography," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 1, pp. 65–73, Jan. 2009.

[16] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers, and K. B. Englehart, "Continuous locomotion-mode identification for prosthetic legs based on neuromuscular–mechanical fusion," *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 10, pp. 2867–2875, Oct. 2011.

[17] M. Goršič *et al.*, "Online phase detection using wearable sensors for walking with a robotic prosthesis," *Sensors*, vol. 14, no. 2, pp. 2776–2794, Feb. 2014.

[18] H. F. Maqbool, M. A. B. Husman, M. I. Awad, A. Abouhossein, N. Iqbal, and A. A. Dehghani-Sanij, "A real-time gait event detection for lower limb prosthesis control and evaluation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 9, pp. 1500–1509, Sep. 2017.

[19] D. Novak *et al.*, "Automated detection of gait initiation and termination using wearable sensors," *Medical Engineering & Physics*, vol. 35, no. 12, pp. 1713–1720, Dec. 2013.

[20] T. R. Clites *et al.*, "Proprioception from a neurally controlled lower-extremity prosthesis," *Science Translational Medicine*, vol. 10, no. 443, p. eaap8373, May 2018.

[21] K. S. Türker, "Electromyography: some methodological problems and issues," *Phys Ther*, vol. 73, no. 10, pp. 698–710, Oct. 1993.

[22] A. E. Patla and J. N. Vickers, "How far ahead do we look when required to step on specific locations in the travel path during locomotion?," *Exp Brain Res*, vol. 148, no. 1, pp. 133–138, Jan. 2003.

[23] S. Wang, H. Pan, C. Zhang, and Y. Tian, "RGB-D image-based detection of stairs, pedestrian crosswalks and traffic signs," *Journal of Visual Communication and Image Representation*, vol. 25, no. 2, pp. 263–272, Feb. 2014.

[24] H.-H. Pham, T.-L. Le, and N. Vuillerme, "Real-Time obstacle detection system in indoor environment for the visually impaired using Microsoft Kinect sensor," *Journal of Sensors*, vol. 2016, Article ID 3754918, 13 pages, 2016.

[25] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: deep learning on point sets for 3d classification and segmentation," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 77–85.

[26] Z. Rozsa and T. Sziranyi, "Obstacle prediction for automated guided vehicles based on point clouds measured by a tilted lidar sensor," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 8, pp. 2708–2720, Aug. 2018.

[27] J.-S. Gutmann, M. Fukuchi, and M. Fujita, "Stair climbing for humanoid robots using stereo vision," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2004, vol. 2, pp. 1407–1413 vol.2.

[28] T. Jung, J. Lim, H. Bae, K. K. Lee, H. Joe, and J. Oh, "Development of the humanoid disaster response platform DRC-HUBO+," *IEEE Transactions on Robotics*, vol. 34, no. 1, pp. 1–17, Feb. 2018.

[29] M. Liu, D. Wang, and H. H. Huang, "Development of an environment-aware locomotion mode recognition system for powered lower limb prostheses," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 4, pp. 434–443, Apr. 2016.

[30] Y. Massalin, M. Abdrakhmanova, and H. A. Varol, "User-independent intent recognition for lower limb prostheses using depth sensing," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 8, pp. 1759–1770, Aug. 2018.

[31] N. E. Krausz, T. Lenzi, and L. J. Hargrove, "Depth sensing for improved control of lower limb prostheses," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 11, pp. 2576–2587, Nov. 2015.

[32] D. C. Ciresan, U. Meier, L. M. Gambardella, and J. Schmidhuber, "Convolutional neural network committees for handwritten character classification," in *2011 International Conference on Document Analysis and Recognition*, 2011, pp. 1135–1139.

[33] H. A. Alwzwazy, H. M. Albehadili, Y. S. Alwan, and N. E. Islam, "Handwritten digit recognition using convolutional neural networks," *International Journal of Innovative Research in Computer and Communication Engineering*, vol. 4, no. 2, pp. 1101–1106, 2016.

[34] D. Xu, Y. Feng, J. Mai, and Q. Wang, "Real-time on-board recognition of continuous locomotion modes for amputees with robotic transtibial prostheses," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 26, no. 10, pp. 2015–2025, Oct. 2018.